

I claim:

Sub
B1

1. A method for accessing memory in a multiprocessor system, the method comprising:

5 from a requesting processor, issuing a request for a block of data to one or more other processors and memory, each copy of the block of data being associated with state information indicating whether the copy is valid or invalid;

in each of the processors and memory that receive the request, checking to determine whether a valid copy of the block of data exists; and

10 returning a valid copy of the requested data from one of the other processors or memory such that only the processor or memory having the valid copy of the data block responds to the request.

Sub
D1

2. The method of claim 1 in which:

15 each of the processors communicates with the memory via a memory controller and each of the processors has a point-to-point link with the memory controller for issuing a request for a block of data to the memory controller.

3. The method of claim 2 in which:

20 each point-to-point link includes two dedicated and unidirectional links.

4. The method of claim 2 in which the point-to-point links are control links for sending and receiving requests for blocks of data.

25 5. The method of claim 2 in which each of the processors has a control path point-to-point link for sending and receiving requests for blocks of data, and a data path point-to-point link for sending and receiving blocks of data.

Sub
C1

30 6. The method of claim 1 in which the processors and shared memory that have an invalid copy of the requested block of data drop the request without responding.

7. The method of claim 1 including:
tracking an identification of a processor that currently has a data block; and
in response to a cache miss in a requesting processor, using the identification to
specifically target a read request to the processor that currently has the requested
5 data block.

8. The method of claim 1 including:
maintaining a directory indicating the one or more processors that have a copy
of a block of data;
10 when the block of data is modified, using the directory to issue a write
invalidation or write update only to the processors that have the copy of the block of
data.

Sub
B2
15 9. A multiprocessor system comprising:
two or more processors, each in communication with a shared memory via a
memory controller;
the processors in communication with the memory controller for issuing a
request for a block of data, each of the processors and the shared memory being
capable of storing a copy of the requested block of data, and each copy of the
20 requested block of data being associated with state indicating whether the copy is
valid or invalid,
each of the processors and the shared memory being responsive to a request
to check itself for a valid copy of a requested block such that only the processor or
shared memory having the valid copy responds to the request for the requested
25 block.

10. The system of claim 9 in which:
each of the processors communicates with the memory via a memory controller
and each of the processors has a point-to-point link with the memory controller for
30 issuing a request for a block of data to the memory controller.

11. The system of claim 10 in which:
each point-to-point link includes two dedicated and unidirectional links.

12. The system of claim 10 in which the point-to-point links are control links for
5 sending and receiving requests for blocks of data.

13. The system of claim 10 in which each of the processors has a control path
point-to-point link for sending and receiving requests for blocks of data, and a data
path point-to-point link for sending and receiving blocks of data.

10

14. The system of claim 9 including:
a directory indicating which processors have a copy of a data block;
wherein the processors are in communication with the directory to identify
which other processors have a copy of the data block, and directing requests for the
15 data block only to processors that have a copy of the data block.

15. The system of claim 14 wherein the directory is incorporated into the data
block.

16. The system of claim 14 wherein the directory is stored in a separate
memory that filters a request and forwards the request only to a processor or
processors that have a copy of the data block.

17. The system of claim 14 wherein the memory controller is in communication
25 with a shared cache, separate from caches of the processors, for buffering most
frequently accessed data blocks.

18. The system of claim 9 wherein each block has state information indicating
which processor currently has a valid copy of a data block, and wherein the
30 processors utilize the state information to specially address a processor having the
valid copy in response to a cache miss in a requesting processor.

Sub
B3

19. A multiprocessor system comprising:
two or more processors, each in communication with a shared memory;
the processors in communication with the shared memory for issuing a request
for a block of data, each of the processors and the shared memory being capable of
5 storing a copy of the requested block of data, and each copy of the requested block
of data being associated with state indicating whether the copy is valid or invalid,
each of the processors and the shared memory being responsive to a request
to check itself for a valid copy of a requested block such that only the processor or
shared memory having the valid copy responds to the request for the requested
10 block.

20. The system of claim 19 wherein each of the processors and the shared
memory is in communication with a control path interconnect, and each of the
processors is in communication with the control path interconnect via a point-to-
15 point link for receiving and sending requests for blocks of data;

each of the processors having a corresponding request queue connecting the
point-to-point link of the processor to the control path interconnect, and each of the
processors having a corresponding snoop queue connecting the point-to-point link of
the processor to the control path interconnect;

20 the request queue in communication with a corresponding processor for
buffering requests for blocks of data by the processor and issuing the requests to
other processors via the control path interconnect; and

the snoop queue in communication with a corresponding processor for buffering
requests for blocks of data destined for the processor.

Add
B4